Data-Driven Linear Koopman Embedding for Networked Systems: Model-Predictive Grid Control

Ramij Raja Hossain, *Graduate Student Member, IEEE*, Rahmat Adesunkanmi, *Graduate Student Member, IEEE*, and Ratnesh Kumar, *Fellow, IEEE*

***Abstract*—This article presents a data-learned linear Koopman embedding of nonlinear networked dynamics and uses it to enable real-time model predictive emergency voltage control in a power grid. The approach involves a novel data-driven "basis-dictionary free" lifting of the system dynamics into a higher dimensional linear space over which a model predictive control is exercised, making it both scalable and rapid for practical real-time implementation. A *Koopman-inspired deep neural network* (KDNN) encoder–decoder architecture for the linear embedding of the underlying networked dynamics under distributive control is presented, in which the end-to-end components of the KDNN, comprising of a triple of transforms is learned from the system trajectory data in one go: a neural network (NN)-based lifting to a higher dimension, a linear dynamics within that higher dimension, and an NN-based projection to the original space. This data-learned approach relieves the burden of the *ad hoc* selection of the nonlinear basis functions (e.g., radial or polynomial) used in conventional approaches for lifting to a higher dimensional linear space. We validate the efficacy and robustness of the approach via application to the standard IEEE 39-bus system.**

***Index Terms*—Data-driven method, Koopman operator, model predictive control (MPC), neural network (NN), voltage control.**

I. Introduction

A. Motivation and Related Works

DESIGNING an efficient control methodology for power grids faces tremendous challenges because of their distributed networked architecture and complex nonlinear dynamics governed by differential-algebraic equations (DAE). Generally, power systems have different layers of controls for different operation modes; among those, emergency controls are of utmost importance for grid security and resilience. Owing to the complexity, the prevalent special protection systems (SPS) for emergency controls are mostly rule-based and easy to implement but limited in their level of resilience. As a result, more recently, researchers have explored optimization-based methods, such as model predictive control (MPC) to develop an analytical approach relying on system dynamics and real-time measurements. But, due to the massive integration of renewable generations, dynamic loads, and inverter-based resources, model complexities and uncertainties are increasing, so having an exact model is difficult. Plus, due to nonlinearity and high-dimensional dynamics, model-based optimizations have high computational burdens, making them time-consuming and not amenable to real-time execution. To this end, the emerging data-driven approaches offer opportunities to control designs in complex networked cyber-physical systems (CPSs) [1], [2]. The data-driven approaches not only avoid the need for knowing the model in advance but also make the control computation fast and real time, trading a small modeling accuracy for a substantial gain in computation speed. Among the data-driven approaches, model-free deep reinforcement learning (DRL) is gaining importance [3], [4]. However, providing analytical guarantees for DRL-based control is still ongoing research and is currently not ready for direct interactions with real-world power grids [5].

Here, we develop a data-driven approach for power systems modeling that lifts the nonlinear dynamics to a higher dimensional linear space and performs the MPC on this linear embedding, making it free of analytical models (that are generally unavailable for complex power grids), scalable, and real-time (for having the control computed in the linear state-space). Also, the lifting is done in a certain NN-based encoder–decoder architecture that avoids the conventional *ad hoc* selection of basis functions (polynomial or radial) for lifting. We first briefly summarize the conventional MPC-based existing works in power systems applications, their computational limitation, and the current lifting approaches for linear embedding and their limitations as well.

1) MPC-Based Approaches in Power Systems: An overview of MPC in power systems can be found in [6]. Quasi-steady-state (QSS) model approximation and sensitivity-based approaches are presented in [7]. MPC formulations with trajectory sensitivity-based approximation of nonlinear DAE are provided in [8], [9], and [10] for emergency voltage control. In [11], a speed-up of trajectory sensitivity computation utilizing the "very dishonest newton (VDHN)" method for the jacobian update is utilized in the MPC context. In recent studies, linear state-space models of power systems are leveraged in MPC-based load frequency control [12], [13], active frequency response [14], and optimal power allocation of energy storage devices [15].

In summary, MPC-based methods are promising for emergency control of power systems, but their real-world application using online optimization over nonlinear DAE faces a major

Manuscript received 31 August 2022; revised 15 December 2022 and 29 January 2023; accepted 27 February 2023. This work was supported in part by the National Science Foundation under Grant NSF-CSSI-2004766 and Grant NSF-PFI-2141084. *(Corresponding author: Ramij Raja Hossain.)*

The authors are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: rhossain@iastate.edu; rahma@iastate.edu; rkumar@iastate.edu).

Digital Object Identifier 10.1109/JSYST.2023.3253041

computational issue, making the MPC inapplicable in real time. This computational bottleneck cannot be overcome even by incorporating different approximation techniques, for instance, sensitivity-based linearization [10].

2) Koopman-Based Lifting and Their Applications in Power Systems: To leverage the scalability of control design in the setting of linear state-space models, Koopman operators have been proposed to provide a higher dimensional linear representation of a nonlinear system [16]. The lifting uses nonlinear basis functions, and the lifted linear dynamics, when projected to the original space, provide an approximation to the original nonlinear dynamics [17], [18]. Finding the suitable basis functions for the lifting operation is a nontrivial problem: in fact, the standard approaches utilize an *ad hocly* predefined dictionary of basis functions (e.g., polynomial or radial) that are not necessarily optimal. The Koopman-based lifting and control design based on it have been explored in power systems applications: [19] solved an MPC-based transient stabilization problem utilizing classical swing dynamics of power systems [20]. A Koopman-based model predictive power system stabilizer (PSS) is designed in [21]. The authors in [22] integrate Koopman-based system identification and controller design for oscillation damping. The Koopman-based controller for decentralized frequency control problem with the time-delayed embedding of measurements is studied in [23]. In [24], data-driven characterization of power system dynamics is investigated using the Koopman operator theory. Overall, the following can be summarized.

1) The Koopman-based embedding in power systems is mostly based on simplified models of power system dynamics so that a predefined dictionary of basis functions (e.g., see [19, eq. (2)]) can be used.
2) Building an appropriate dictionary of basis functions for complex power system dynamics comprising higher-order generator models, exciter models, load models, etc., is challenging, and this restricts the application of Koopman-based methods in power system control design. This issue has also been observed in other complex nonlinear applications.
3) In recent research [25], [26], [27], [28], [29], [30], [31], DNN-based basis functions have been explored mostly in non-networked systems. Within the power systems (an example of a networked system), only [32] considers learning-based lifting for transient stability problems, but based on a simplified model of the swing dynamics (similar to the simplified model used in [19]).

B. Our Approach and Key Contributions

Motivated by these observations, we present a Koopman-inspired deep neural network architecture, termed KDNN, for a linear embedding of the nonlinear voltage dynamics, by drawing data from the system evolution based on its complete differential-algebraic dynamics (without any approximation) and design an MPC-based emergency voltage control over the linear embedding. The approach presented is applicable to any networked system. The proposed KDNN employs an encode–decoder architecture, embedding an interim linear stage. The validity of the proposed KDNN-based MPC scheme is established by implementing an emergency voltage recovery scheme following a severe fault in the IEEE 39-bus system. The performance and robustness of the scheme are demonstrated via $\pm 20\%$ load variations around the nominal model as well as under different contingencies. In summary, the key contributions of the work presented include the following.

1) A Koopman-inspired linear embedding to map the nonlinear voltage dynamics to a lifted linear state-space for the emergency voltage control in power systems, considering the higher order generator models, exciter models, load models, etc. (unlike the previous works which used the simplified swing dynamics), and also allowing the recovery of bus voltages through control actions (e.g., reactive power compensation).
   Our approach is "*basis-dictionary free*" unlike the standard approaches that pick *ad hoc* basis functions (polynomial or radial) that are not necessarily optimal. To this end, we formulate an encoder–decoder architecture comprising of the following:
   a) NN-based data-learned basis functions for lifting;
   b) data-learned linear Koopman operators associated with the lifted states and control inputs;
   c) data-learned projection function used for mapping down to the original state space.
   This encoder–decoder-based neural architecture is trained in an end-to-end manner *in one go*, learning directly from data the unknown basis function, the linear Koopman operators, and the projection function together. Such end-to-end learning also eliminates the risks of numerical issues associated with iterative learning of neural basis functions versus the least-square estimation of Koopman operators, as is the case in [32].
2) We provided a detailed *noise robustness analysis* of the neural lifting by providing bounds on the lifted states given the bounds on the original voltage states, meaning that by design, the lifting is bounded input bounded output stable.
3) The MPC computation for emergency voltage control is directly performed in the lifted linear state space, which makes the MPC *scalable as well as real time* for power system application: We demonstrate a 36-fold speed-up of control computation on the IEEE 39-bus system.

II. Formulation of Proposed Methodology

This section starts by considering the nonlinear power system dynamics representation in the DAE form that implicitly models the bus voltage dynamics under corresponding controls. We then briefly overview the Koopman operators and present the proposed data-driven KDNN architecture for embedding the nonlinear voltage dynamics into a higher dimensional linear space. Finally, we formulate the MPC framework in the lifted linear space to stabilize voltage trajectories following any disturbances.

A. Power System Model

The dynamics of a power system can be captured in the form of a nonlinear DAE, given as follows:

$$\dot{X} = F(X, Y, U); \quad 0 = G(X, Y, U) \tag{1}$$

where $X :=$ state variables, $Y :=$ algebraic variables, and $U :=$ control inputs. Here, we are interested in the voltage stability control, so the relevant dynamics is the response of bus voltages $V \subseteq Y$ to the control input U (reactive VAR compensation in our setting). Note that the DAEs (1) are an interacting system of simultaneous equations written compactly employing vectors: its state equations F are corresponding to the state variables, whereas the vector of all the network constraints involving voltages, active, and reactive powers form the algebraic equations G.

As noted previously, the use of the nonlinear model (1) is not scalable for real-time MPC-based voltage stabilization, and instead, we aim to learn the relevant part of the dynamics, namely the controlled voltage dynamics, utilizing the data obtained from the DAE model trajectories in the proposed KDNN framework. Since the system trajectory data are available in discrete time, we examine the voltage dynamics in discrete time, indexed by integer variable $k \geq 0$, as

$$V_{k+1} = \mathcal{T}(V_k, U_k) \tag{2}$$

where $V_k := [V_k^1, \ldots, V_k^n]^T$ is the voltage vector and $U_k := [U_k^1, \ldots, U_k^m]^T$ is the control vector at kth control instant in a n-bus power grid with m number of control inputs. The underlying voltage dynamics, as a function of controls distributed over the entire network, in vector form is given as (2), whose dimension is the number of network nodes.

In a practical setting, the control implementation occurs at a larger time period T_c, compared to the voltage discretization time period, T_s. We let $H = T_c/T_s$ denote the length of voltage trajectory history between any two control instants. The controls are held constant between the two control instances, while the voltage dynamics continue to evolve, forming a time series of length H between any two control instants. Thus, at each control instant k, the "voltage-state" is taken to be the time-series of voltage values between the control instants $k-1$ and k, that can be captured in a matrix form: $V_k^{\text{hist}} = [V_{k,j}^i]$, where $i = 1, \ldots, n$, and $j = 0, \ldots, H-1$. In contrast, the control input U_k takes a new value only at the instant k. Henceforth, with a slight abuse of notation, we use V_k to represent V_k^{hist}.

Next, we discuss the lifting of the nonlinear mapping $\mathcal{T}(\cdot,\cdot)$ of (2) to a higher dimensional linear state space by way of employing the theory of Koopman operators.

B. Preliminaries on the Koopman Operator

Let us consider discrete-time nonlinear dynamics as

$$x_{k+1} = f(x_k, u_k) \tag{3}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, and f is the vector field. We adopt the lifting mechanism presented in [18], which only lifts the state variables x to a higher dimension, leaving the control variables u unlifted. According to the Koopman operator theory for finite-dimensional linear approximation of the nonlinear system defined by (3), there exist $N >> n$, real-valued nonlinear basis functions (lifting functions) $\mathcal{G}_i : \mathbb{R}^n \to \mathbb{R}$ for $i = 1, \ldots, N$ forming $\mathcal{G} = [\mathcal{G}_1, \ldots, \mathcal{G}_N]^\top$, which lifts the original state space to the higher dimensional state-space so that in the lifted space, the system follows the linear dynamics

$$\mathcal{G}(x_{k+1}) = \mathcal{A}\,\mathcal{G}(x_k) + \mathcal{B}\,u_k \tag{4}$$

where $\mathcal{A} \in \mathbb{R}^{N\times N}$ and $\mathcal{B} \in \mathbb{R}^{N\times m}$ are the Koopman operators associated with the state and control spaces, respectively.

One way to obtain the Koopman operators for a finite-dimensional approximation such as (4) is to utilize the method of extended dynamic mode decomposition (EDMD) [17], [18], where a predefined dictionary of basis functions is assumed known. The approach requires collecting T tuples $\{(x_k, u_k, x_{k+1})_{k=1}^T\}$ from various trajectory data of the given nonlinear system under different initial conditions. The collected trajectory data are arranged as: $\mathbf{X} = [x_1, \ldots, x_T], \mathbf{X}^+ = [x_2, \ldots, x_{T+1}], \mathbf{U} = [u_1, \ldots, u_T]$, where the ith elements $\mathbf{X}(i), \mathbf{X}^+(i)$, and $\mathbf{U}(i)$ satisfy the dynamics (3), namely, $\mathbf{X}^+(i) = f(\mathbf{X}(i), \mathbf{U}(i))$. For a given dictionary of mappings: $\bar{\mathcal{G}}(x) = \begin{bmatrix}\bar{\mathcal{G}}_1(x) & \cdots & \bar{\mathcal{G}}_N(x)\end{bmatrix}^\top$, the lifted space state representations are: $\mathbf{X}_{\text{lift}} = \begin{bmatrix}\bar{\mathcal{G}}(x_1) & \cdots & \bar{\mathcal{G}}(x_T)\end{bmatrix}$, and $\mathbf{X}^+_{\text{lift}} = \begin{bmatrix}\bar{\mathcal{G}}(x_2) & \cdots & \bar{\mathcal{G}}(x_{T+1})\end{bmatrix}$. Then, the matrices $\mathcal{A}$ and $\mathcal{B}$ can be obtained by solving a least-square optimization problem

$$\min_{\mathcal{A},\mathcal{B}} ||\mathbf{X}^+_{\text{lift}} - \mathcal{A}\mathbf{X}_{\text{lift}} - \mathcal{B}\mathbf{U}||_{\text{F}} \tag{5}$$

where $\|\cdot\|_{\text{F}}$ denotes the Frobenius norm of a matrix. The solution of (5) gives

$$[\mathcal{A}\ \ \mathcal{B}] = \mathbf{X}^+_{\text{lift}}[\mathbf{X}_{\text{lift}}\ \ \mathbf{U}]^\dagger \tag{6}$$

where $\dagger$ denotes the Moore–Penrose pseudo-inverse of a matrix. In standard EDMD formulation, the projection from the higher dimensional space to the original lower dimensional space is achieved by $\mathcal{C} \in \mathbb{R}^{n\times N}$, computed by solving

$$\min_{\mathcal{C}} ||\mathbf{X} - \mathcal{C}\mathbf{X}_{\text{lift}}||_{\text{F}} \tag{7}$$

which yields $\mathcal{C} = \mathbf{X}\mathbf{X}^\dagger_{\text{lift}}$. The computation of (6) is sometimes prohibitive for large datasets ($T >> N$), in which case it is beneficial to instead solve a "normal form" equation obtained by multiplying $\mathcal{V} := \begin{bmatrix}\mathbf{X}_{\text{lift}}\\ \mathbf{U}\end{bmatrix}\begin{bmatrix}\mathbf{X}_{\text{lift}}\\ \mathbf{U}\end{bmatrix}^\top$ on both sides of (6) to obtain $[\mathcal{A}\ \ \mathcal{B}]\mathcal{V} = \mathcal{W}$, where $\mathcal{W} := \mathbf{X}^+_{\text{lift}}\begin{bmatrix}\mathbf{X}_{\text{lift}}\\ \mathbf{U}\end{bmatrix}^\top$. Then, $[\mathcal{A}\ \ \mathcal{B}] = \mathcal{W}\mathcal{V}^\dagger$, in which the dimension of $\mathcal{W}$ and $\mathcal{V}$ are $N \times (N+m)$, and $(N+m)\times(N+m)$, respectively, and are independent of the number of data samples T. Analogously, we can solve the normal form associated with $\mathcal{C} = \mathbf{X}\mathbf{X}^\dagger_{\text{lift}}$ to get $\mathcal{C} = \widehat{\mathcal{W}}\widehat{\mathcal{V}}^\dagger$, with $\widehat{\mathcal{W}} = \mathbf{X}\mathbf{X}^\top_{\text{lift}}$, and $\widehat{\mathcal{V}} = \mathbf{X}_{\text{lift}}\mathbf{X}^\top_{\text{lift}}$ having dimensions $n \times N$ and $N \times N$, respectively, that are again independent of T.

C. Proposed NN-Based Koopman Embedding

Instead of relying on any predefined dictionary of basis function choices, here we explore the power of artificial intelligence (AI) in data-driven learning of the basis functions, represented as deep neural networks (DNNs), as in [27]. Viewing the linear embedding to be a special case of a DNN, our approach learns not only the lifting functions (mapping to a higher dimension) but also the linear embedding (in the higher dimension), as well

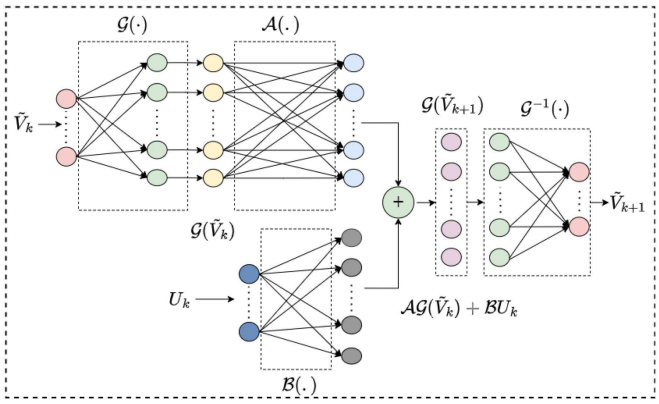


Fig. 1. End-to-end KDNN architecture.

as the projections (mapping to the original lower dimension), all three in one go.

Following the Koopman operator theory that approximates (3) as (4), we can write the voltage dynamics (2) in the lifted linear space as

$$\mathcal{G}(\tilde{V}_{k+1}) = \mathcal{A}\,\mathcal{G}(\tilde{V}_k) \;+\; \mathcal{B}\,U_k \tag{8}$$

where $\tilde{V}_k = V_k - V_{\text{ref}}$. Note that this implies that at the reference point, $\tilde{V}_k = 0$.

Our encoder–decoder KDNN architecture to realize (8) is presented in Fig. 1, where the matrix structure of $\tilde{V}_k$ (same as V_k) of dimension $[n, H]$ is flattened to form an vector input of length $\overline{n} = n \times H$.

1) Construction of the Encoder: The flattened vector input $\tilde{V}_k$ is passed through a fully connected neural network (FCNN) layer (feed-forward structure) with N neurons and element-wise nonlinear activation function $\tanh(\cdot)$. This transformation lifts the actual state, voltage-series $\tilde{V}_k$, to a higher dimensional state $Z_k = \mathcal{G}(\tilde{V}_k)$, where $\mathcal{G}(\cdot)$ represents the nonlinear function associated with the NN-based encoder operation.

2) Linear Embedding: Once lifted to a suitable higher dimension, the linear transformation of $\mathcal{G}(\tilde{V}_k)$ by the Koopman operator $\mathcal{A}(\cdot)$ is an instance of another FCNN (with identity activation function) to be trained as shown in Fig. 1. Similarly, the input U_k is also processed through the Koopman operator $\mathcal{B}(\cdot)$, which is yet another instance of an FCNN with an identity activation function (see Fig. 1). The output of these two operations is added in the lifted space to obtain $\mathcal{A}\,\mathcal{G}(\tilde{V}_k) + \mathcal{B}\,U_k$, which equals $\mathcal{G}(\tilde{V}_{k+1})$ as per (8).

3) Construction of the Decoder: Finally, to project to the original state-space, we utilize a decoder architecture $\mathcal{G}^{-1}(\cdot)$ shown in Fig. 1. The decoder, an FCNN layer with $\overline{n}$ neurons and element-wise nonlinear activation function $\tanh(\cdot)$ denoted by $\mathcal{G}^{-1}(\cdot)$, does the inverse operation of the encoder $\mathcal{G}(\cdot)$, and takes $\mathcal{G}(\tilde{V}_{k+1}) = \mathcal{A}\,\mathcal{G}(\tilde{V}_k) + \mathcal{B}\,U_k$ as input and maps it to predicted $\tilde{V}_{k+1}^{\text{pred}}$.

4) Loss Function for End-to-End Training: The end-to-end KDNN is parameterized by θ, and can be expressed as $\mathcal{F}(\cdot, \theta)$.

We formulate a loss function based on four mappings, that makes the trained KDNN robust to errors.

1) $\tilde{V}_{k+1}^{\text{pred},(a)}$ obtained from $\{\tilde{V}_k, U_k\}$ through (8), and decoder mapping $\mathcal{G}^{-1}(\cdot)$ is compared against ground-truth $\tilde{V}_{k+1}$.
2) The encoder value $\mathcal{G}(\tilde{V}_k)$ is passed through $\mathcal{G}^{-1}(\cdot)$ to obtain the estimated version $\tilde{V}_k^{\text{pred},(b)}$, which is compared against the ground-truth $\tilde{V}_k$.
3) $\tilde{V}_{k+1}$ is passed through the encoder $\mathcal{G}(\cdot)$ and decoder $\mathcal{G}^{-1}(\cdot)$ to obtain the estimated version $\tilde{V}_{k+1}^{\text{pred},(c)}$, which is compared against the ground-truth $\tilde{V}_{k+1}$.
4) $\mathcal{G}(\tilde{V}_{k+1})^{(a)}$ obtained using (8) in step (1), and $\mathcal{G}(\tilde{V}_{k+1})^{(c)}$ obtained by passing $\tilde{V}_{k+1}$ through the encoder $\mathcal{G}(\cdot)$ in step (3).

Accordingly, the mean squared error loss function is formulated as follows:

$$\mathcal{L}(\theta) = \frac{1}{L}\sum_i \Bigg[\left(\tilde{V}_{k+1} - \tilde{V}_{k+1}^{\text{pred},(a)}\right)_i^2 + \left(\tilde{V}_k - \tilde{V}_k^{\text{pred},(b)}\right)_i^2 + \left(\tilde{V}_{k+1} - \tilde{V}_{k+1}^{\text{pred},(c)}\right)_i^2 + \left\{\mathcal{G}(\tilde{V}_{k+1})^{(a)} - \mathcal{G}(\tilde{V}_{k+1})^{(c)}\right\}_i^2\Bigg] \tag{9}$$

where L denotes the batch size and i denotes the ith sample of a single minibatch. We used minibatch stochastic gradient descent to train $\mathcal{F}(\cdot, \theta)$.

D. Formulation of MPC Problem

Here, the objective of the control design is to minimize the postdisturbance voltage deviations with respect to a user-defined reference value $V_{\text{ref}} = 1.00$ p.u. For this, we employ MPC, which, at each control instant, computes an optimal sequence of control inputs for the remaining control horizon by optimizing a predicted future behavior of the underlying system, implements the first control action of the computed sequence, and then, repeats the same process with the new measurements at the next control instant [10]. This iterative control computation, employing new state measurements, helps to correct the effects of modeling error as compensated by the new measurement taken. Using the trained KDNN architecture, we can extract the functions $\mathcal{G}(\cdot), \mathcal{A}(\cdot)$, and $\mathcal{B}(\cdot)$ that are then utilized in the computation of the MPC in the lifted linear space as formulated next.

The measured voltages V_k are lifted to $Z_k := \mathcal{G}(\tilde{V}_k) = \mathcal{G}(V_k - V_{\text{ref}})$. Note that at $V_k = V_{\text{ref}}$, $\tilde{V}_k = 0$, which implies that $Z_{\text{ref}} = \mathcal{G}(V - V_{\text{ref}}) = \tanh(0) = 0$, for zero bias in neural lifting (see Section III). Accordingly, at any control instant k, the model-predictive optimization problem in the lifted space is an instance of linear quadratic regulator as follows:

$$\min_{U_k,\ldots,U_{k+N_k-1}} \sum_{i=0}^{N_k-1} \left[Z_{k+i+1}^\top\, Q\, Z_{k+i+1} + U_{k+i}^\top\, R\, U_{k+i}\right] \tag{10a}$$

$$\text{s.t.} \quad Z_{k+i+1} = \mathcal{A}\, Z_{k+i} + \mathcal{B}\, U_{k+i} \quad \forall i \in [0, N_k - 1] \tag{10b}$$

$$U_{\min} \le U_{k+i} \le U_{\max} \quad \forall i \in [0, N_k - 1] \tag{10c}$$

$$Z_{k+N_k} \in \mathcal{Z}_{\text{terminal}} \tag{10d}$$

where N_k is the number of remaining control instants at the instant k, $Q \in \mathbb{R}^{N\times N}$ and $R \in \mathbb{R}^{m\times m}$ are the user-specified weight matrices (with N being the dimension of lifted space, and m being the dimension of the control space), $U_{\max}$ and $U_{\min}$ are the lower/upper control bounds, and $\mathcal{Z}_{\text{terminal}}$ is the desired final set in the lifted space, as detailed in Section III. Showing stability of the proposed MPC then boils down to showing that $\mathcal{Z}_{\text{terminal}}$ is contained in the region of attraction of the equilibrium under the computed MPC. On the other hand, for the feasibility of the MPC, some states in $\mathcal{Z}_{\text{terminal}}$ must be reachable from the initial state. The computation of the region of attraction and that of forward reachability can be found in existing literature such as [33], [34], [35], and [36].

III. Properties of Neural Lifting

A. Terminal Constraints and Tracking in Lifted Space

We first show that the terminal constraints in the original space can be mapped to constraints in the lifted space.

Proposition 1. (Terminal Constraints): The terminal constraints on voltage, namely, $\tilde{V}_{\min} \le \tilde{V}_i \le \tilde{V}_{\max}$ can be mapped to a corresponding constraint set $\mathcal{Z}_{\text{terminal}}$ in the lifted space, under positive weights and zero bias in the neural lifting.

Proof: Noting that the lifting function corresponds to a single-layered fully connected feed-forward network, given a voltage V, or equivalently, $\tilde{V} = V - V_{\text{ref}}$, the lifted state $Z = \mathcal{G}(\tilde{V}) \in \mathbf{R}^N$ can be expressed as

$$Z = \phi(W\tilde{V} + b) \equiv \tanh(W\tilde{V} + b) \tag{11}$$

where $W \in \mathbf{R}^{N\times\overline{n}}$ is the weight matrix, $b \in \mathbf{R}^N$ is the bias vector, and $\phi(\cdot) \equiv \tanh(\cdot)$ is the activation function.

Under the positivity of the entries of the weight matrix $W = [W_{ij}]$, namely, $W_{ij} \ge 0\,\forall\, i, j$, and zero bias, $b = 0$, given the element-wise bound on voltage, $\tilde{V}_{\min} \le \tilde{V}_i \le \tilde{V}_{\max}\ \forall i = 1, \ldots, \overline{n}$, we can obtain element-wise bound on $Z_j = \tanh(\sum_{i=1}^{\overline{n}} W_{j,i}\tilde{V}_i)$ by employing the monotonicity of $\tanh(\cdot)$ as

$$Z_{j,\min} \le Z_j \le Z_{j,\max} \quad \forall j = 1, \ldots, N \tag{12}$$

where $Z_{j,\min} := \tanh(\sum_{i=1}^{\overline{n}} W_{j,i}\tilde{V}_{\min})$ and $Z_{j,\max} := \tanh(\sum_{i=1}^{\overline{n}} W_{j,i}\tilde{V}_{\max})$. These element-wise bounds $[Z_{j,\min}, Z_{j,\max}]\ \forall j$ yield the terminal constraint set $\mathcal{Z}_{\text{terminal}}$, as desired. ■

Remark 1: Reference [37] notes that the $\tanh(\cdot)$ activation has a slope that is sector-bounded in [0, 1], and that can be expressed as a quadratic constraint (QC). Using this, it determines a QC bound on Z given the bound on V. However, this requires solving a semidefinite programming in a higher dimensional lifted space that is numerically challenging. To mitigate this issue, we have provided previously the element-wise simpler bounds for the lifted states.

The next proposition justifies the reference tracking in the lifted space, by showing that it implies reference tracking also in the original space. Recall that the objective of the standard nonlinear MPC is to minimize the deviation of the predicted trajectory from the desired reference in the original space. Proposition 2 shows that the encoder function $\mathcal{G}(\cdot)$ (with zero bias) preserves the closeness to reference, thereby justifying the tracking in the lifted space [as in (10a)].

Proposition 2: For any $\epsilon > 0$, $||Z||_2 \le \epsilon$ implies that there exists $\delta > 0$ such that $||V - V_{\text{ref}}||_2 \le \delta$. Conversely for any $\epsilon' > 0$, $||V - V_{\text{ref}}||_2 \le \epsilon'$ implies the existence of $\delta' > 0$ such that $||Z||_2 \le \delta'$.

Proof: We start by proving the first statement. $||Z||_2 \le \epsilon$, implies that $\sum_i Z^2(i) \le \epsilon^2$ (here $Z(i)$ denotes the ith element of Z), which implies $\forall i, Z(i)^2 \le \epsilon^2 \Rightarrow (\tanh((WV - WV_{\text{ref}})(i)))^2 \le \epsilon^2 \Rightarrow |(\tanh((WV - WV_{\text{ref}})(i))| \le \epsilon$. Due to symmetry, $|\tanh(\cdot)| = \tanh(|\cdot|)$. Hence, $|\tanh((WV - WV_{\text{ref}})(i))| \le \epsilon \Rightarrow \tanh|(WV - WV_{\text{ref}})(i)| \le \epsilon \Rightarrow |(WV - WV_{\text{ref}})(i)| \le \tanh^{-1}(\epsilon)$, where the last implication follows from the monotonicity and non-decreasing property of $\tanh(\cdot)$. Taking squared sum over all elements, we have $\sum_i |(WV - WV_{\text{ref}})(i)|^2 \le N \times (\tanh^{-1}(\epsilon))^2$ (recall N is the dimension of Z), which is equivalent to $||WV - WV_{\text{ref}}||_2 \le \sqrt{N \times (\tanh^{-1}(\epsilon))^2}$. Finally, from the property of norm, we have $||V_- V_{\text{ref}}||_2 \le \frac{\sqrt{N\times(\tanh^{-1}(\epsilon))^2}}{\sigma_{\min}(W)} := \delta > 0$, where the smallest singular value $\sigma_{\min}(W) > 0$ for a full rank W. This proves the first part; the second part can be established in a similar manner, and it can be shown that $\delta' = \sqrt{N \times (\tanh((\epsilon'||W||_2)))^2}$, where we need to utilize the fact that $\|W(V - V_{\text{ref}})\|_2 \le \|W\|_2 \cdot \|V - V_{\text{ref}}\|_2$. ■

It follows from Proposition 2 that the reference tracking in the original voltage space can be mapped to the reference tracking in the lifted space.

B. Noise/Disturbance Robustness of Lifting Operation

Considering the safety-critical nature of power systems, certifying the robustness of neural lifting against input uncertainties is important. In this regard, we show here the boundedness of the perturbation in the lifted space whenever the perturbation due to noise or input disturbance in the original space is bounded.

Proposition 3. (Noise-Robustness): For a bounded noise perturbation of voltage inputs, namely when $\forall i \in \overline{n} : V_{i,\min} \le \tilde{V}_i \le V_{i,\max}$, and the control bounds $[U_{\min}, U_{\max}]$ on the input U, the perturbation in the predicted states of the lifted space is also bounded under positive weights and zero bias in the neural encoding.

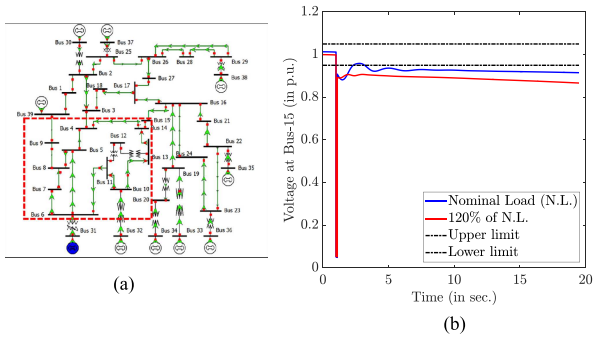
Proof: Following the derivation of (12) in Proposition 1, we have: $Z_{j,\min} \le Z_j \le Z_{j,\max}\ \ \forall j = 1, \ldots, N$, where $Z_{j,\min} := \tanh(\sum_{i=1}^{\overline{n}} W_{j,i}V_{i,\min})$ and $Z_{j,\max} := \tanh(\sum_{i=1}^{\overline{n}} W_{j,i}V_{i,\max})$.

Additionally, with the positivity of $\mathcal{A} \in \mathbf{R}^{N\times N}$, with $\mathcal{A}_{ij} \ge 0\forall\, i, j$ and $\mathcal{B} \in \mathbf{R}^{N\times m}$, with $\mathcal{B}_{ij} \ge 0\,\forall\, i, j$, and the disturbance bounds $[U_{\min}, U_{\max}]$ on the control input U, the predicted states in lifted space $Z^+ = \mathcal{A}Z + \mathcal{B}U$ are bounded as

$$\mathcal{A}\, Z_{\min} + \mathcal{B}\, U_{\min} \le Z^+ \le \mathcal{A}\, Z_{\max} + \mathcal{B}\, U_{\max} \tag{13}$$

where $Z_{\min} := [Z_{0,\min}, \ldots, Z_{N,\min}]^\top$ and $Z_{\max} := [Z_{0,\max}, \ldots, Z_{N,\max}]^\top$. ■

It follows from Proposition 3, that the effect of noise or disturbance remains bounded when we move from the original nonlinear space to the lifted linear space, i.e., the proposed neural encoding is robust to noise and disturbance.



(a) (b)

Fig. 2. (a) IEEE 39-bus system. (b) Voltage plots for fault with no control.

IV. Test Cases, Implementation, and Results

This section presents the proof-of-validation of our proposed methodology with the IEEE 39-bus system. We start with the description of the test system, followed by the main results.

A. Test System Description

We consider a modified version of the standard IEEE 39-bus system to be able to study the emergency voltage control problems. The IEEE 39-bus system consists of ten generators from bus-30 to bus-39 and 46 transmission lines. The generators have Park-Concordia fourth-order state-space model with states rotor angle δ, speed ω, q-axis transient voltage e'_q and d-axis transient voltage e'_d. Except for the slack one at bus 39, the generators are equipped with IEEE Type-1 excitation systems as mentioned in [38] and [39]. The parameter values of transmission lines, generators, excitation systems, base-load, and generation levels are taken from [38]. More details on the generators and exciter model can be found in the power system analysis toolbox (PSAT) manual [39]. The load dynamics are modeled using exponential recovery load [39], [40] for which load recovery time constants $T_P = T_Q = 30$, steady-state load exponents $\alpha_s = 0.1$, $\alpha_t = 2$, $\beta_s = 2.8$, and transient load exponents $\beta_t = 4$ are selected according to [10], with certain modifications to increase the criticality of load recovery. The system is simulated using the open-source simulator PSAT [39].

For our study, we selected a subnetwork of 12 load buses $\mathcal{S} = \{4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$ [the red dotted region in Fig. 2(a)]. It is observed that after a fault at bus 15, which gets cleared by tripping a line between buses 15 and 16 within six cycles of system operation, voltages in the vicinity of the fault bus drop below the desired level almost immediately [see Fig. 2(b)] and without any corrective control this may cause a system collapse. Similar observations are also made for faults at other buses, for instance, buses 4, 7, and 8; the situation gets exacerbated under increased load conditions and limited generator excitation. This necessitates an emergency control mechanism, and we implemented our proposed MPC-based voltage stabilization policy following faults to keep the voltage trajectories close to a reference value $V_{\text{ref}} = 1.00$ p.u. The control inputs are reactive power compensation at buses 4, 5, 7, 8, 10, 12, 14, and 15 that can provide fast reactive support of up to 0.25 p.u. in each control step. To stabilize the voltage trajectories, we choose a five-step MPC with a control horizon of 15 s, divided into five segments of 3 s each, i.e., $T_c = 3$.

B. Training of KDNN

1) Training Data: For training the proposed KDNN, we created a large pool of training data by simulating the system under various operating conditions. As given in (1), the nominal model of a power system follows DAE dynamics, plus the nominal system is subject to unknown load fluctuations and faults, and we need to ensure that the training data capture those changes in operating conditions. To capture the dependencies on changing conditions, we introduce a parameter Θ extending the DAE to $\dot{X} = F(X, Y, U, \Theta), 0 = G(X, Y, U, \Theta)$. In the training data generation phase, random variations of control inputs U in the range [0,0.25] p.u. were introduced, and further, the parametric changes in Θ were introduced by way of load variations of $\pm 20\%$ around the nominal loads plus by applying three different contingencies. For each T_f sec simulation ($T_f \approx 20$ s for the IEEE-39 Bus system), we randomly chose a Θ value, and randomly chose controls at each of the five control instants (i.e., Θ, namely the load and the contingency, were held constant in each short time period of T_f sec simulation). We repeated this procedure of trajectory generation multiple times. Finally, all these data were pooled together and used for the training of KDNN, ensuring that the KDNN is exposed to different operating conditions, thereby making it suitable for the dynamic variations of the power systems. We also show the applicability of our method by applying random load levels within $\pm 20\%$ of nominal level, and five different contingencies. In total, we utilized 1000 random load conditions and faults at buses 15, 4, and 7, collecting a total of $1000 \times 5 \times 3 = 15000$ data samples in the form of $\{(V_k, U_k, V_{k+1})\}$ triples. We divided the data into $70:30$ to create the training and test datasets.

2) KDNN Architecture: As shown in Fig. 1, the lifting function $\mathcal{G}(\cdot)$ is built using an FCNN layer, where the number of neurons of the FCNN layer decides the dimension N of the lifted space. We considered a subnetwork of 12 buses, which implies $n = 12$, and observation history length $H = 4$ between any $(k-1)$th to kth control instants (this is because the sampling rate for voltage values is higher than the control application rate, and so there are multiple voltage values between any two control values). We tested $N = 256 > 5 \times n \times H$. The activation function of the FCNN layer is $\tanh(\cdot)$ representing the nonlinearity of the layer $\mathcal{G}(\cdot)$. The linear transformation matrices $\mathcal{A}$ and $\mathcal{B}$ are also represented as single-layer FCNNs with 256 neurons, while their activation function is identity.

3) Training Parameters and Data Preprocessing: The optimizer chosen for the training is ADAM, with gradient momentum $\beta_1 = 0.9/0.95$ and rms momentum $\beta_2 = 0.999/0.95$. The loss function, batch size, learning rate, and performance metric are: mean squared error (MSE) loss, 32, 10^{-3}, and R^2-score, respectively. Prior to training the KDNN, the input and output data were processed as follows: First, $V_{\text{ref}} = 1.00$ was subtracted from the voltage values V_k to obtain the corresponding $\tilde{V}_k$ values. Next, this adjusted data were normalized in the range $[-1, 1]$ preserving $\tilde{V}_k = 0$ at $V_k = V_{\text{ref}}$. The control values U_k in the [0,0.25] range were also normalized in the $[-1, 1]$ range.

4) Training and Testing Performance: We utilized Tesnorflow to build and train the KDNN model. The training is done by minimizing a composite loss function of four components mentioned in Section II-C. To ensure bounded-input bounded-output

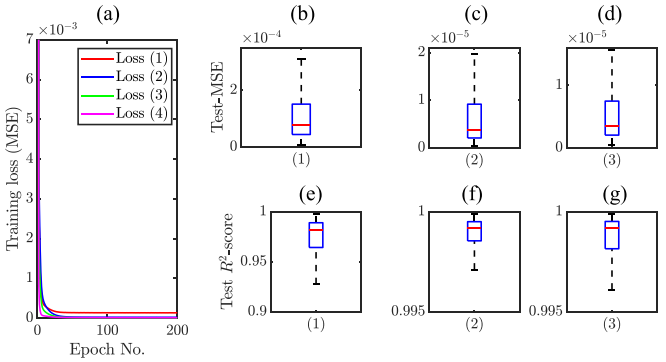


Fig. 3. (a) KDNN training loss. (b)–(d) KDNN test MSE for $\tilde{V}_{k+1}^{\text{pred},(a)}$, $\tilde{V}_{k}^{\text{pred},(b)}$, and $\tilde{V}_{k+1}^{\text{pred},(c)}$, respectively. (e)–(g) KDNN test R^2-score for $\tilde{V}_{k+1}^{\text{pred},(a)}$, $\tilde{V}_{k}^{\text{pred},(b)}$, and $\tilde{V}_{k+1}^{\text{pred},(c)}$, respectively.

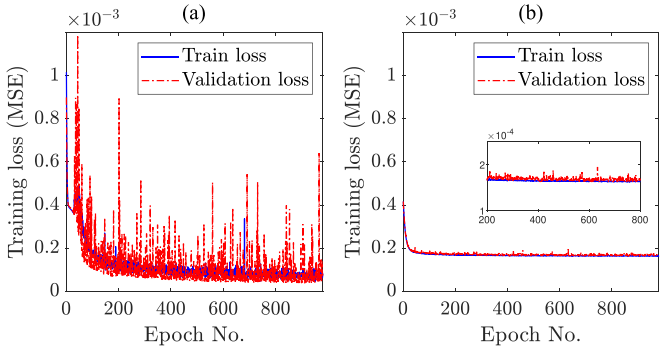


Fig. 4. KDNN training. (a) Without preprocessing. (b) With preprocessing.

property of the architecture for robustness to noise and other perturbations, the training is performed ensuring the positivity of weight matrices as well as matrices of the linear embedding (as discussed and elaborated in Sections II and III). The training losses in MSE are shown in Fig. 3(a).

To show the generalization property of our trained model, we evaluated the model performance on testing data (which was not used during training). For reconstruction of $\tilde{V}_{k+1}^{\text{pred},(a)}$, $\tilde{V}_{k}^{\text{pred},(b)}$, and $\tilde{V}_{k+1}^{\text{pred},(c)}$ using testing data and the trained model, we evaluated the performance in terms of MSE and the coefficient of determination $R^2 \in [0, 1]$ (recall an R^2 value of 1 indicates an exact fit). The plots given in Fig. 3(b)–(d) for the test MSE and Fig. 3(e)–(g) for the R^2-score demonstrate that the generalization performance of the trained model is satisfactory. Next noting that both training and testing performances are satisfactory, we utilized the trained KDNN to extract the functions $\mathcal{G}(\cdot), \mathcal{A}(\cdot)$, and $\mathcal{B}(\cdot)$ for MPC computation in the lifted linear space.

5) Discussion on Model Training, Generalization Error, and Overfitting: The proposed KDNN learns the mapping of the voltage dynamics from the kth control instant voltage V_k and control U_k to the next control instant voltage, $V_{k+1} = \mathcal{T}(V_k, U_k)$ under different operating conditions of the power system. For the training of the KDNN, we carried out certain key steps to ensure a good generalization performance.

1) We varied the load condition to within $\pm 20\%$ of the nominal load condition, applied contingencies, and employed random policies to execute control actions, storing the triples, $\{V_k, U_k, V_{k+1}\}$. This provides enough richness to the training data for reducing the generalization error.
2) We used V_k, which is a time series of voltage values in-between the $(k-1)$th and kth control instants, rather a single voltage value as training data.
3) The loss function involved four different ways of measuring error to provide multiple cross checks as presented in Section II-C, (9).
4) Finally, to avoid scaling-related issues due to different voltage levels at different buses, we subtracted the voltage values with respect to a common reference point V_{ref} and normalized all data values in $[-1, 1]$, respectively.

We validated the effectiveness of data preprocessing of step (4) by performing cross validation, dividing the available training dataset randomly into train data (80%) and validation data (20%) during model training. The plots in Fig. 4 show that the variance in validation loss is lower in the case of training with the data preprocessing of step (4). Thus, preprocessing of dataset with step (4) further helped us reduce the fitting error. Also, as seen in Fig. 4(b), the training loss and validation loss closely matched, showing no sign of overfitting.

C. MPC Computation at the Embedded Linear Space

We computed the MPC-based controls to minimize the voltage deviations from a reference $V_{\text{ref}} = 1.00$ p.u. The design parameters are $Q = I_{N\times N}$, while $R = 0$, as the controls are reactive power (VAR) compensation, it does not incur any effective control cost for the power utility like load shedding (which curtails active loads). Since the steady-state voltage needs to be in the range of [0.95, 1.05] p.u., in the MPC formulation for the final lifted state Z_{k+N_k}, we introduced the corresponding terminal constraint Z_{terminal} with $V_{\text{terminal}} := [0.95, 1.05]$ and $V_{\text{ref}} = 1$, as detailed in Section III-A. For validating the robustness of the proposed control design, (a) 5 different load levels (80%, 90%, 100%, 110%, and 120% of the nominal load (N.L.)), and (b) 8 different fault buses (bus-15, bus-4, bus-7, bus-5, bus-6, bus-8, bus-10, bus-14) were selected. Note, all these faults were cleared by tripping respective transmission lines, resulting in different underlying network topologies. The MPC computations were done in the linear embedded state space, solving the constrained optimization (10). Next, the computed controls were applied to the original nonlinear system. The voltage profiles for each of the aforementioned cases are shown in Figs. 5–8, validating that the proposed scheme successfully achieved the desired voltage performance under different operating conditions, thereby confirming the effectiveness and robustness of the proposed methodology of designing controls using the KDNN-based lifted linear embedding of the nonlinear dynamics. Also, note that the voltage values at $t \approx 20$ are within the safe limits: $V_{\text{min}} = 0.95$ p.u. and $V_{\text{max}} = 1.05$ p.u.

We also computed the control actions at each control instants and plotted the accumulated control actions for the five different load cases in Figs. 9–12. The trend suggests that with the

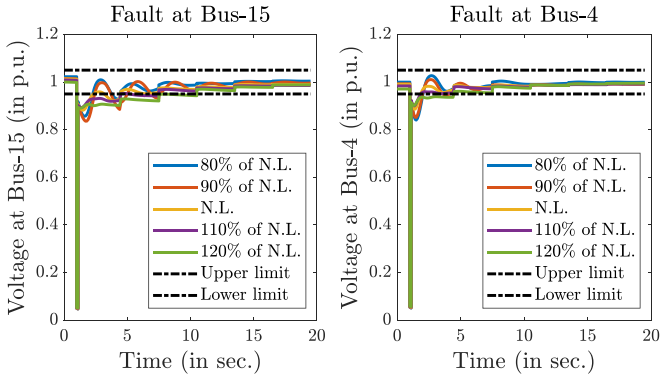


Fig. 5. Voltage plots for faults at Bus-15 and Bus-4.

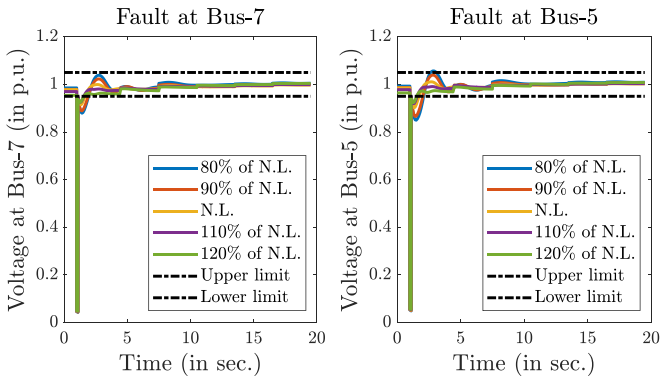


Fig. 6. Voltage plots for faults at Bus-7 and Bus-5.

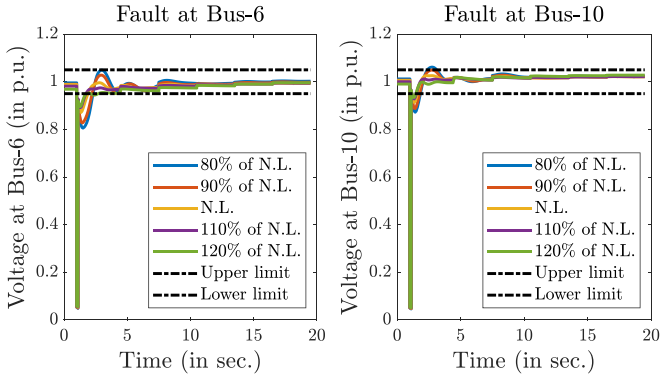


Fig. 7. Voltage plots for faults at Bus-6 and Bus-10.

load increase, the amount of VAR compensation (reactive power compensation) increased, and this trend is expected.

D. Discussion on Noise Robustness

Section III-B provides the theoretical understanding of noise robustness for the lifted states, and due to the high dimensional problem structure, it is difficult to represent bounds on lifted state variables. To validate this property, we carried out the MPC computation considering that the measurement voltages are affected by a Gaussian noise of mean 0, and standard deviation 1%. The voltage and control plots for (a) three random load

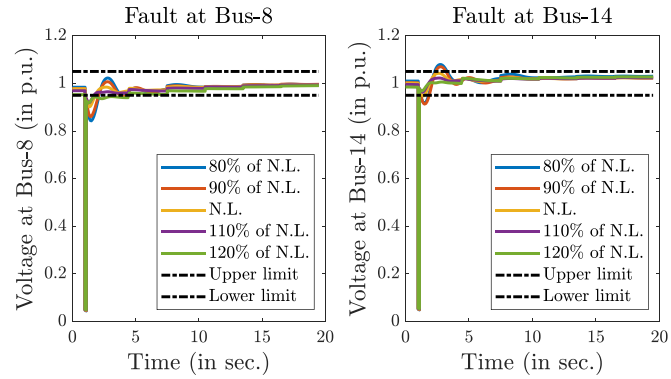


Fig. 8. Voltage plots for faults at Bus-8 and Bus-14.

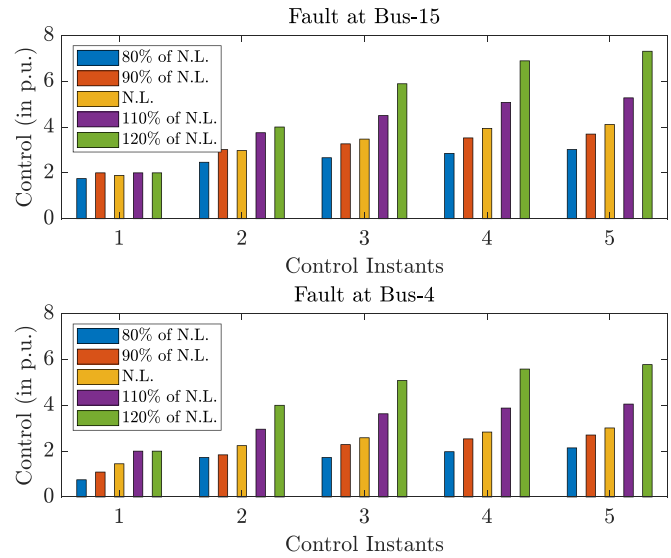


Fig. 9. Cumulative control plots for faults at Bus-15 and Bus-4.

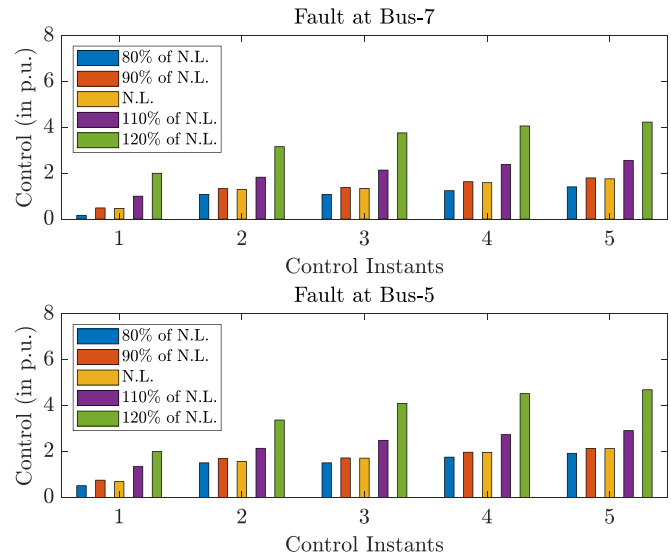


Fig. 10. Cumulative control plots for faults at Bus-7 and Bus-5.

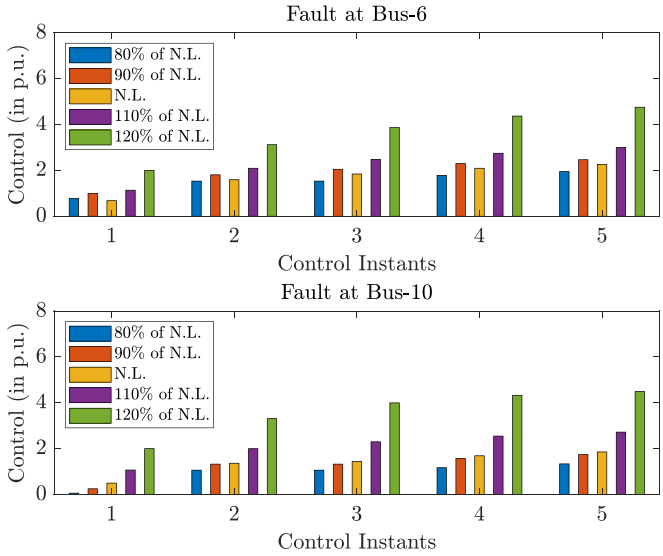


Fig. 11. Cumulative control plots for faults at Bus-6 and Bus-10.

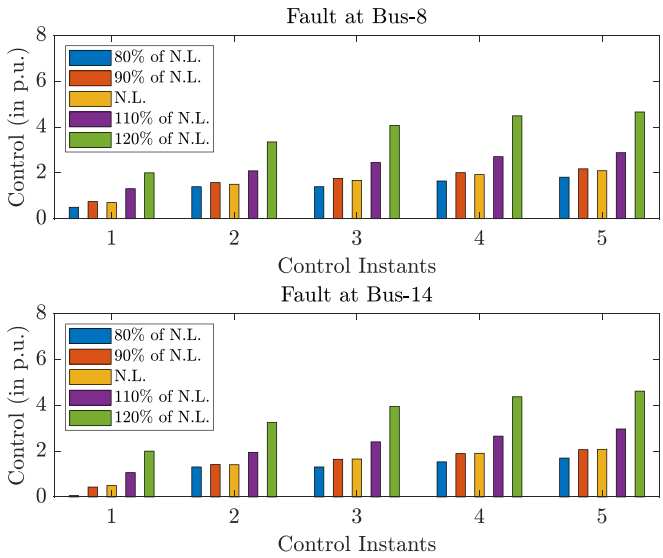


Fig. 12. Cumulative control plots for faults at Bus-8 and Bus-14.

levels (within $\pm 20\%$ of nominal load) and (b) faults at bus-15 and 6 are shown in Figs. 13 and 14, respectively.

E. Comparison With Standard Approaches

1) Prediction Performance: Standard Koopman Versus KDNN: As mentioned in [27], an early work in an NN-based Koopman design, radial basis function, polynomials, and kernel functions are mostly common basis functions in the Koopman-operator-based control design. But the choice of an appropriate basis function is an open problem, and this is our primary motivation for utilizing NNs to use data to learn the appropriate basis functions. Here, we demonstrate the benefit of the proposed KDNN-based design compared to the standard EDMD method mentioned in Section II-B for approximating nonlinear implicit voltage dynamics. First, note that for the polynomial basis, the

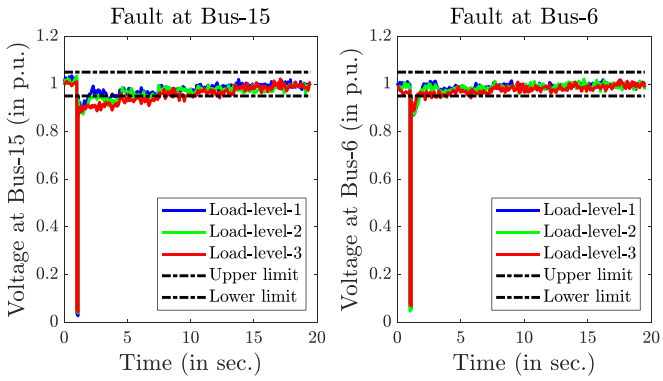


Fig. 13. Voltage plots for faults at Bus-15 and Bus-6.

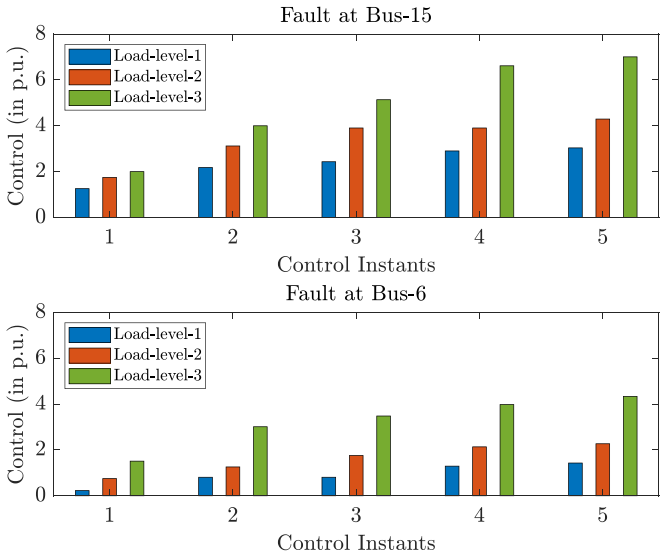


Fig. 14. Cumulative control plots for faults at Bus-15 and Bus-6.

choice of candidate basis set increases double-exponentially with the dimension of original nonlinear dynamics, $n \times H$, as per the problem formulation. This makes creating an optimal choice of dictionary nonscalable and hence prohibitive. So, for comparison, we picked radial basis functions with a dictionary size of 2000. The centers of the radial basis functions were determined by K-means clustering over training data, while for the spread parameter, we selected $\sigma = 0.05$. We then followed (5)–(7) to find $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ over the same training data utilized to train the KDNN. Next, the computed matrices are used to find the predicted values of the voltage dynamics over the same testing data used for the KDNN testing. The mean squared errors (MSEs) of both predictions and R^2-score are plotted in Fig. 15, which clearly shows the superiority of our KDNN-based learned basis functions.

2) Traditional MPC Versus KDNN MPC: We compared the performance of our proposed method with the state-of-art MPC [10] for voltage recovery as well as control computation time. Both traditional MPC [10] and our proposed method uses the same MATLAB-based PSAT platform for the MPC implementation. What differs is that while [10] employs a

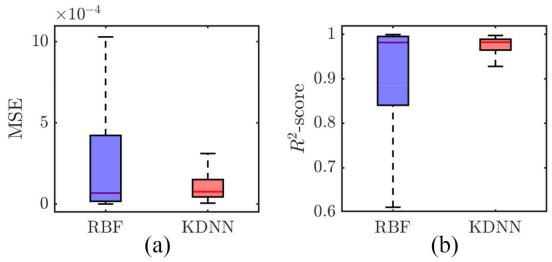


Fig. 15. Comparison of standard lifting and proposed lifting method.

TABLE I
COMPARISON OF PERFORMANCE FOR THE IEEE-39 BUS SYSTEM

Scenarios	Performance measure ($\mathcal{J}$)	
	Traditional MPC [10]	KDNN MPC
Fault at Bus-15	7.5769	7.6106
Fault at Bus-4	12.4760	12.5362
Fault at Bus-7	10.2480	10.3150
Fault at Bus-5	13.5311	13.6136
Fault at Bus-6	15.2707	15.4424
Fault at Bus-10	12.8511	12.8845
Fault at Bus-14	11.5472	11.6870
Fault at Bus-8	10.5985	10.6714

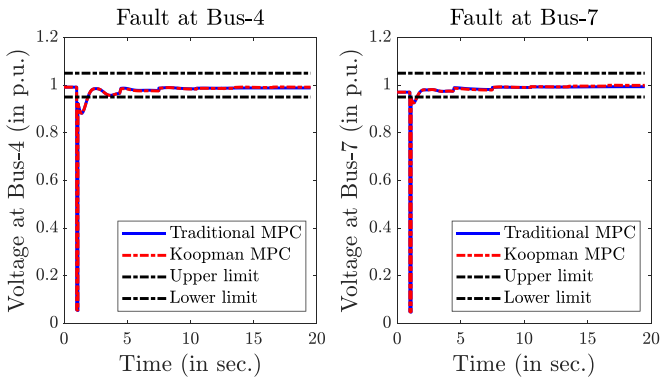


Fig. 16. Comparison of voltage plots for faults at Bus-4 and Bus-7.

voltage trajectory sensitivity-based method that necessitates time-consuming dynamic simulation and additional MATLAB routines to compute the trajectory sensitivities, no such trajectory sensitivity computation is needed in the proposed setting as it solves only a linear MPC and that in fact results in the speed-up of the overall MPC computation. One can define a *performance measure* $\mathcal{J}$ as an aggregation of the squared sum of the voltage trajectory deviations of the buses in $\mathcal{S}$ with respect to the reference ($V_{\text{ref}} = 1.00$ p.u.). We applied different fault conditions under random load levels, and the resulting performance measure $\mathcal{J}$ as noted in Table I turns out to be almost the same (maximum percentage error being $\approx 1.2\%$) for both the traditional MPC [10] and the proposed KDNN MPC. A representative voltage plot to show the similarity is given in Fig. 16.

The contrast between the control computation time for trajectory-sensitivity-based state-of-art MPC [10] and that of the proposed KDNN-based MPC is shown in Table II. The KDNN-based approach shows an impressive 36-fold speed-up of the control computation time. Also, the proposed method takes 0.18 s to compute a control at each online decision instant, making MPC real-time and practical for power systems for a first time. It is important to note that even the traditional controllers, e.g., UVLS relaying scheme, generally needs ∼0.5 s to decide a control action [41]. We used standard Intel(R) Xeon(R) CPU E3-1240 v6 at 3.70-GHz processor with 16-GB RAM for our implementation and computation.

TABLE II
COMPARISON OF COMPUTATION TIME FOR THE IEEE-39 BUS SYSTEM

Method	Average time / MPC step
Traditional MPC in [10]	6.50 sec
KDNN MPC	0.18 sec

F. Discussion on Real-World Control Room Application

The training data for KDNN are obtained in a simulated environment. The use of simulation is standard practice for different offline studies within the power utility control room, e.g., state estimation, contingency analysis, and dynamic security analysis [42]. Moreover, with recent progress on test-bed simulators, digital twins are also adopted to match the simulation with real-time operation [43]. For the application of the proposed method, the control room operators need to set a nominal model of the system and generate offline training data for KDNN training as mentioned in Sections IV-A and IV-B; train the KDNN architecture offline, and finally, during the online deployment phase; and (c) collect the voltage measurements and solve the optimization problem (10) using the functions $\mathcal{G}(\cdot), \mathcal{A}(\cdot)$, and $\mathcal{B}(\cdot)$ extracted from the trained KDNN.

Gaps between the simulation models and the real-world systems may exist, and in case over time, if the underlying power system dynamics differ greatly from its existing nominal model (because of major operational changes/modifications), then the existing nominal model itself must be updated by repeating the offline simulation of the model and the training of the KDNN.

V. CONCLUSION

This article implemented a Koopman-inspired encoder–decoder framework for the data-driven linear embedding of networked systems dynamics, paving the way for designing a control strategy in the lifted linear state-space, making the MPC design scalable and real-time for power grids. We combined the power of deep learning with the Koopman operator theory for lifting nonlinear dynamics into a higher dimensional linear space. Our data-driven approach autolearns the basis/projection functions removing the burden of selecting those arbitrarily, traditionally taken to be polynomials or radial bases. The test results applied to the IEEE 39-bus system validated the proposed scheme's performance in terms of efficacy, robustness against load variations and fault conditions, and further a 36-fold computational speed-up making MPC real time. We also validated the superiority of our approach compared to the standard EDMD approaches for Koopman embedding, which employs predefined basis functions. The proposed promising technique of unraveling the implicit nonlinear dynamics, combining Koopman theory and deep learning methods, opens up a new direction of control design for complex nonlinear networked systems.

REFERENCES

[1] P. Radanliev, D. De Roure, M. Van Kleek, O. Santos, and U. Ani, "Artificial intelligence in cyber physical systems," *AI Soc.*, vol. 36, no. 3, pp. 783–796, 2021.

[2] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality, and robustness," *IEEE Trans. Autom. Control*, vol. 65, no. 3, pp. 909–924, Mar. 2020.

[3] Z. Zhang, D. Zhang, and R. C. Qiu, "Deep reinforcement learning for power system applications: An overview," *CSEE J. Power Energy Syst.*, vol. 6, no. 1, pp. 213–225, 2020.

[4] Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and Z. Huang, "Adaptive power system emergency control using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1171–1182, Mar. 2020.

[5] X. Chen, G. Qu, Y. Tang, S. Low, and N. Li, "Reinforcement learning for selective key applications in power systems: Recent advances and future challenges," *IEEE Trans. Smart Grid*, vol. 13, no. 4, pp. 2935–2958, Jul. 2022.

[6] I. A. Hiskens and M. Vrakopoulou, *Model Predictive Control for Power Networks*. Berlin, Germany: Springer, 2021, pp. 1234–1239.

[7] M. Glavic and T. van Cutsem, "Some reflections on model predictive control of transmission voltages," in *Proc. 38th North Amer. Power Symp.*, 2006, pp. 625–632.

[8] M. Zima and G. Andersson, "Model predictive control employing trajectory sensitivities for power systems applications," in *Proc. IEEE Conf. Decis. Control*, 2005, pp. 4452–4456.

[9] I. Hiskens and B. Gong, "MPC-based load shedding for voltage stability enhancement," in *Proc. IEEE 44th Conf. Decis. Control*, 2005, pp. 4463–4468.

[10] L. Jin, R. Kumar, and N. Elia, "Model predictive control-based real-time power system protection schemes," *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 988–998, May 2010.

[11] G. Hou and V. Vittal, "Trajectory sensitivity based preventive control of voltage instability considering load uncertainties," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 2280–2288, Nov. 2012.

[12] J. Liu, Q. Yao, and Y. Hu, "Model predictive control for load frequency of hybrid power system with wind power and thermal power," *Energy*, vol. 172, pp. 555–565, 2019.

[13] A. Oshnoei, M. Kheradmandi, R. Khezri, and A. Mahmoudi, "Robust model predictive control of gate-controlled series capacitor for LFC of power systems," *IEEE Trans. Ind. Inform.*, vol. 17, no. 7, pp. 4766–4776, Jul. 2021.

[14] C. Jin, W. Li, J. Shen, P. Li, L. Liu, and K. Wen, "Active frequency response based on model predictive control for bulk power system," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3002–3013, Jul. 2019.

[15] R. K. Subroto, K. L. Lian, C.-C. Chu, and C.-J. Liao, "A fast frequency control based on model predictive control taking into account of optimal allocation of power from the energy storage system," *IEEE Trans. Power Del.*, vol. 36, no. 4, pp. 2467–2478, Aug. 2021.

[16] B. O. Koopman, "Hamiltonian systems and transformation in hilbert space," *Proc. Nat. Acad. Sci. United States Amer.*, vol. 17, no. 5, pp. 315–318, 1931.

[17] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the Koopman operator: Extending dynamic mode decomposition," *J. Nonlinear Sci.*, vol. 25, no. 6, pp. 1307–1346, 2015.

[18] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[19] M. Korda, Y. Susuki, and I. Mezić, "Power grid transient stabilization using Koopman model predictive control," *IFAC-PapersOnLine*, vol. 51, no. 28, pp. 297–302, 2018.

[20] P. Kundur, N. J. Balu, and M. G. Lauby, *Power System Stability and Control*. vol. 7. New York, NY, USA: McGraw-hill, 1994.

[21] W. Han and A. M. Stankovic, "Koopman model predictive control-based power system stabilizer design," in *Proc. 52nd North Amer. Power Symp.*, 2021, pp. 1–6.

[22] W. Han and A. M. Stanković, "Model-predictive control design for power system oscillation damping via excitation—A data-driven approach," *IEEE Trans. Power Syst.*, vol. 38, no. 2, pp. 1176–1188, Mar. 2023.

[23] X. Li, C. Mishra, S. Chen, Y. Wang, and J. De La Ree, "Determination of parameters of time-delayed embedding algorithm using Koopman operator-based model predictive frequency control," *CSEE J. Power Energy Syst.*, vol. 7, no. 6, pp. 1140–1151, 2021.

[24] P. Sharma, V. Ajjarapu, and U. Vaidya, "Data-driven identification of nonlinear power system dynamics using output-only measurements," *IEEE Trans. Power Syst.*, vol. 37, no. 5, pp. 3458–3468, Sep. 2022.

[25] E. Yeung, S. Kundu, and N. Hodas, "Learning deep neural network representations for Koopman operators of nonlinear dynamical systems," in *Proc. Amer. Control Conf.*, 2019, pp. 4832–4839.

[26] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature Commun.*, vol. 9, no. 1, pp. 1–10, 2018.

[27] Y. Han, W. Hao, and U. Vaidya, "Deep learning of Koopman representation for control," in *Proc. IEEE Conf. Decis. Control*, 2020, pp. 1890–1895.

[28] X. Wang, Y. Kang, and Y. Cao, "Deep Koopman operator based model predictive control for nonlinear robotics systems," in *Proc. Int. Conf. Adv. Robot. Mechatronics*, 2021, pp. 931–936.

[29] R. Wang, Y. Han, and U. Vaidya, "Deep Koopman data-driven optimal control framework for autonomous racing," to be published, doi: 10.13140/RG.2.2.21512.75526.

[30] Y. Xiao, X. Zhang, X. Xu, X. Liu, and J. Liu, "Deep neural networks with Koopman operators for modeling and control of autonomous vehicles," *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 135–146, Jan. 2023.

[31] H. Shi and M.Q.-H. Meng, "Deep Koopman operator with control for nonlinear systems," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 7700–7707, Jul. 2022.

[32] Z. Ping, Z. Yin, X. Li, Y. Liu, and T. Yang, "Deep Koopman model predictive control for enhancing transient stability in power grids," *Int. J. Robust Nonlinear Control*, vol. 31, no. 6, pp. 1964–1978, 2021.

[33] A. El-Guindy, D. Han, and M. Althoff, "Estimating the region of attraction via forward reachable sets," in *Proc. Amer. Control Conf.*, 2017, pp. 1263–1270.

[34] B. Schürmann, N. Kochdumper, and M. Althoff, "Reachset model predictive control for disturbed nonlinear systems," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 3463–3470.

[35] L. Jin, R. Kumar, and N. Elia, "Reachability analysis based transient stability design in power systems," *Int. J. Elect. Power Energy Syst.*, vol. 32, no. 7, pp. 782–787, 2010.

[36] S. Talukder, M. Ibrahim, and R. Kumar, "Resilience indices for power/cyberphysical systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 51, no. 4, pp. 2159–2172, Apr. 2021.

[37] M. Fazlyab, M. Morari, and G. J. Pappas, "Safety verification and robustness analysis of neural networks via quadratic constraints and semidefinite programming," *IEEE Trans. Autom. Control*, vol. 67, no. 1, pp. 1–15, Jan. 2022.

[38] M. A. Pai, *Energy Function Analysis for Power System Stability*. Berlin, Germany: Springer, 1989.

[39] F. Milano, "An open source power system analysis toolbox," *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1199–1206, Aug. 2005.

[40] B.-K. Choi et al., "Measurement-based dynamic load models: Derivation, comparison, and validation," *IEEE Trans. Power Syst.*, vol. 21, no. 3, pp. 1276–1283, Aug. 2006.

[41] Y. Dong, X. Xie, K. Wang, B. Zhou, and Q. Jiang, "An emergency-demand-response based under speed load shedding scheme to improve short-term voltage stability," *IEEE Trans. Power Syst.*, vol. 32, no. 5, pp. 3726–3735, Sep. 2017.

[42] C. Brosinsky, D. Westermann, and R. Krebs, "Recent and prospective developments in power system control centers: Adapting the digital twin technology for application in power system control centers," in *Proc. IEEE Int. Energy Conf.*, 2018, pp. 1–6.

[43] F. Arraño-Vargas and G. Konstantinou, "Modular design and real-time simulators toward power system digital twins implementation," *IEEE Trans. Ind. Inform.*, vol. 19, no. 1, pp. 52–61, Jan. 2023.

Ramij Raja Hossain (Graduate Student Member, IEEE) received the B.E degree in electrical engineering from Jadavpur University, Kolkata, India, in 2013. He is currently working toward the Ph.D. degree in electrical engineering with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA.

He served as a Senior Engineer, Distribution with CESC Limited, India, from 2013 to 2018. His current research includes data-driven control, distributed optimization, and artificial-intelligence-based approaches for security, stability, and control of power systems.

Rahmat Adesunkanmi (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from the University of Ibadan, Ibadan, Nigeria, in 2015. She is currently working toward the Ph.D. degree in electrical engineering with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA.

Her research interest includes noise-robust machine learning, and different clustering techniques.

Ms. Adesunkanmi is an active Member of the Graduate students Society for Women in Engineering.

Ratnesh Kumar (Fellow, IEEE) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology Kanpur, Kanpur, India, in 1987 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas at Austin, Austin, TX, USA, in 1989 and 1991, respectively.

He is a Palmer Professor with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA, where he directs the Embedded Software, Sensors, Networks, Cyberphysical, and Energy (ESSeNCE) Lab. Previously, he held faculty position with the University of Kentucky, and various visiting positions with the University of Maryland (College Park), the Applied Research Laboratory at the Pennsylvania State University (State College), the NASA Ames, the Idaho National Laboratory, the United Technologies Research Center, and the Air Force Research Laboratory.

Dr. Kumar is a Fellow of the American Association for the Advancement of Science, and was a Distinguished Lecturer of the IEEE Control Systems Society. He was the recipient of D. R. Boylan Eminent Faculty Award for Research and Award for Outstanding Achievement in Research from Iowa State University, and also the Distinguished Alumni Award from IIT Kanpur. He was the recipient of Gold Medals for the Best EE Undergrad, the Best All Rounder, and the Best EE Project from IIT Kanpur, and the Best Dissertation Award from UT Austin, the Best Paper Award from the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, and has been Keynote Speaker and paper award recipient from multiple conferences. He is or has been an Editor of several journals (including of IEEE, SIAM, ACM, Springer, IET, MDPI).